ธาวินทร์ ลีลาคุณารักษ์

การจัดการฐานข้อมูลแบบ Access ด้วย Delphi

เพื่อให้เห็นภาพการจัดการฐานข้อมูลแบบ Access ด้วย Delphi

ใช่ว่าท่านผู้อ่านที่สนใจทางด้านคอมพิวเตอร์ คงจะรู้จักฐานข้อมูลแบบ Access เป็นอย่างดี เนื่องจากโปรแกรม MS Access เป็นโปรแกรมบริหารฐานข้อมูลแบบ Relational Database ระดับพื้นฐาน ซึ่งหน่วยงานต่าง ๆ ที่มีชุดไมโครซอฟต์ออฟฟิศใช้งานส่วนใหญ่จะมีโปรแกรม MS Access อยู่ในมือ และข้อมูลที่จำเป็นต่อการดำเนินงานของหน่วยงานบางส่วนหรืออาจทั้งหมดก็จะถูกจัดเก็บไว้ในฐานข้อมูลแบบ Access ดังกล่าวนี้ด้วย

สำหรับ Delphi เองก็ถือว่าเป็นเครื่องมือพัฒนาโปรแกรมที่ทำงานบนไมโครซอฟต์วินโดวส์ที่กำลังได้รับความสนใจมากขึ้นตามลำดับ และมีหลายองค์กรได้จัดหา Delphi มาพัฒนาโปรแกรมของตนบ้างแล้ว ซึ่งถ้าหากหน่วยงานนั้น ๆ จัดเก็บข้อมูลต่าง ๆ ไว้ในฐานข้อมูลแบบ Access และคุ้นเคยกับการแก้ปัญหาต่าง ๆ ที่เกิดขึ้นกับระบบฐานข้อมูลดังกล่าวอยู่แล้ว

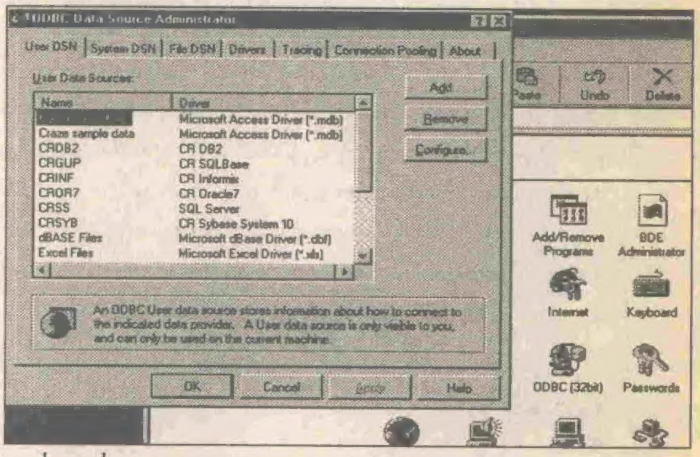
เมื่อมีการนำ Delphi เข้ามาใช้ในการพัฒนาโปรแกรม ส่วนใหญ่จะไม่ต้องการสร้างระบบฐานข้อมูลแบบใหม่ขึ้น เนื่องจากไม่ต้องการเสียเวลาในการจัดทำข้อมูลซ้ำอีก และไม่ต้องการเสียเวลาในการสร้างความคุ้นเคยกับฐานข้อมูลแบบใหม่ที่จะต้องนำมาใช้เก็บข้อมูลต่าง ๆ แทน แต่กิจการที่เคยใช้ MS Access อยู่ก่อนหน้านั้นส่วนใหญ่จะต้องการให้โปรแกรมที่สร้างด้วย Delphi สามารถจัดการกับฐานข้อมูลแบบ Access ที่มีอยู่เดิมได้โดยตรง

การเชื่อมต่อฐานข้อมูลแบบ Access กับ Delphi

โดยทั่วไปผลิตภัณฑ์ทั้งของไมโครซอฟท์และอินไพรส์ (บอร์แลนด์) จะมีส่วนกลไก (Engine) ที่ใช้จัดการกับระบบฐานข้อมูลของตัวเอง สำหรับ MS Access จะจัดการกับระบบฐานข้อมูลของตน (.mdb) โดยผ่านกลไกที่เรียกว่า "Microsoft Jet Database Engine" ส่วนโปรแกรมใช้งานที่สร้างบน Delphi จะจัดการกับระบบฐานข้อมูลของตน โดยสร้างผ่านส่วนกลไกที่เรียกว่า "Borland Database Engine" ซึ่งกลไกการจัดการฐานข้อมูลทั้งสองรายนี้จะมีวิธีการดำเนินกิจกรรมกับระบบฐานข้อมูลที่แตกต่างกัน

ถึงแม้ทั้ง MS Access และ Delphi จะใช้กลไกจัดการฐานข้อมูลคนละระบบ แต่ในการสร้างโปรแกรมด้วย Delphi ให้สามารถจัดการกับฐานข้อมูลแบบ Access ได้นั้น จะมีขั้นตอนไม่ยุ่งยากมากนัก เนื่องจากไมโครซอฟท์เจ้าของระบบปฏิบัติการวินโดวส์ที่มีคนใช้มากที่สุด ได้เตรียมเครื่องมือจัดการฐานข้อมูลที่เรียกว่า Open Database Connectivity (ODBC) ซึ่งเครื่องมือชุดนี้จะทำหน้าที่เชื่อมต่อระบบจัดการฐานข้อมูลที่แตกต่างกันให้สามารถทำงานร่วมกันได้

ถ้าจะสรุปให้ง่ายขึ้นก็คือ การที่โปรแกรมคอมพิวเตอร์ที่สร้างด้วย Delphi จะสามารถจัดการกับระบบฐานข้อมูลแบบ Access ได้นั้น โปรแกรมดังกล่าวจะต้องดำเนินการผ่านเครื่องมือหรือกลไกจัดการระบบฐานข้อมูลถึง 3 ชุดด้วยกันคือ ดำเนินการผ่าน Borland Database Engine (BDE), Open Database Connectivity และ Microsoft Jet Database Engine ตามลำดับ แต่อย่างไรก็ดี ในแง่ของผู้ใช้และผู้พัฒนาโปรแกรมจะไม่เห็นถึงความซับซ้อนของการจัดการกับฐานข้อมูลในลักษณะดังกล่าวมากนัก เนื่องจากกลไกจัดการฐานข้อมูลต่าง ๆ ในข้างต้นจะทำการติดต่อกันเองโดยอัตโนมัติหลังเชื่อมโยงความสัมพันธ์เข้าด้วยกันแล้ว และเวลาที่ใช้ในการดำเนินการกับฐานข้อมูลของกลไกในแต่ละส่วนจะอยู่ในระดับที่ยอมรับได้ (ทดลองบนเครื่องคอมพิวเตอร์ที่ใช้ CPU ระดับ 486)



รูปที่ 1 เครื่องมือ ODBC

ขั้นตอนการเชื่อมความสัมพันธ์ระหว่าง Access กับ Delphi

โดยปกติโปรแกรมใช้งานทั่วไปที่สร้างด้วย Delphi จะไม่สามารถจัดการกับระบบฐานข้อมูลแบบ Access ได้โดยตรง เราจำเป็นต้องทำการเชื่อมโยงความสัมพันธ์ระหว่างฐานข้อมูลแบบ Access กับ Delphi เข้าด้วยกันก่อน โดยหลังจากที่เราสร้างฐานข้อมูลขึ้นบน Access แล้ว (หรืออาจใช้ฐานข้อมูลที่มีอยู่เดิม) เราจะใช้เครื่องมือ ODBC ที่ไมโครซอฟท์จัดเตรียมไว้ให้เป็นตัวกลางในการเชื่อมโยงความสัมพันธ์ระหว่างตารางข้อมูลแบบ Access กับ Delphi ซึ่งลำดับขั้นตอนมีดังนี้

1. เรียก ODBC ขึ้นมาทำงาน โดยเข้าไปที่ Control Panel ของวินโดวส์ แล้วใช้เมาส์ดับเบิลคลิกที่ไอคอนของ ODBC ตามลำดับ

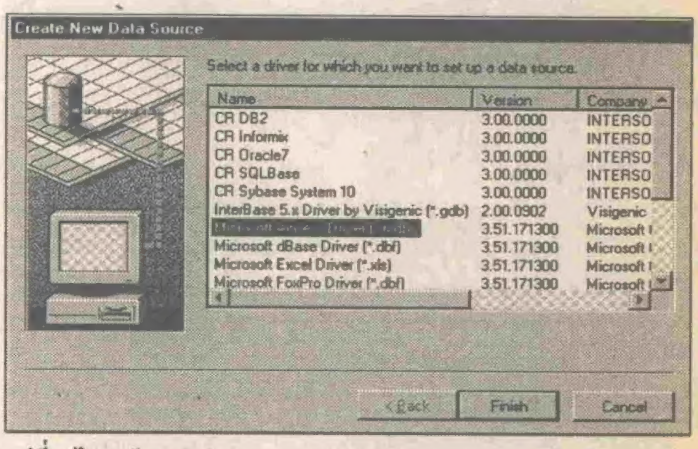
2. ใช้เมาส์คลิกที่ปุ่มคำสั่ง "Add" ของเครื่องมือ ODBC เพื่อเรียกไดอะล็อก Create New Data Source ขึ้นมาทำงาน

3. เลือกคลิกที่หัวข้อ "Microsoft Access Driver" บนไดอะล็อก Create New Data Source แล้วคลิกที่ปุ่มคำสั่ง "Finish" ตามลำดับ

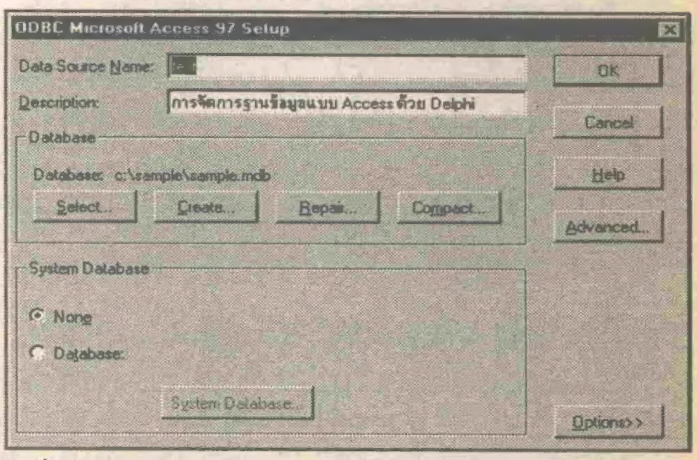
4. ใส่ชื่อฐานข้อมูลลงในช่อง Data Source Name บนไดอะล็อก ODBC Microsoft Access Setup ที่ปรากฏขึ้น โดยชื่อฐานข้อมูลดังกล่าวจะเป็นชื่อเรียกฐานข้อมูลที่ Delphi มองเห็น โดยทั่วไปเราจะตั้งชื่อฐานข้อมูลที่สื่อความหมายชัดเจน เช่น ถ้าเป็นฐานข้อมูลเกี่ยวกับคลังสินค้าจะกำหนดชื่อเป็น "Store" หรือ "Stock Control" หรือถ้าเป็นฐานข้อมูลที่เกี่ยวกับการจัดการในซูเปอร์มาร์เก็ต จะกำหนดชื่อเป็น "Supermarket" เป็นต้น

5. ใช้เมาส์คลิกที่ปุ่มคำสั่ง "Select" ของไดอะล็อก ODBC Microsoft Access Setup เพื่อเลือกแฟ้มฐานข้อมูลที่จะนำมาใช้งานจากฮาร์ดดิสก์ (เลือกจากไดอะล็อก Select Database)

6. ใช้เมาส์คลิกที่ปุ่มคำสั่ง OK ของเครื่องมือ ODBC เพื่อปิดโปรแกรม ซึ่งหมายถึงการเชื่อมโยงฐานข้อมูลแบบ Access กับ Delphi เสร็จสมบูรณ์แล้ว



รูปที่ 2 ไดอะล็อก Create New Data Source



รูปที่ 3 ไดอะล็อก ODBC Microsoft Access Setup

โดยปกติถ้าเราไม่เชื่อมความสัมพันธ์ระหว่างฐานข้อมูลแบบ Access กับ Delphi ไว้ เมื่อเราไปเรียกโปรแกรม Database Explorer ที่มากับ Delphi ขึ้นมาทำงานบนจอภาพ โปรแกรมดังกล่าวจะมองไม่เห็นฐานข้อมูลแบบ Access ที่จะใช้ประกอบการสร้างโปรแกรม แต่หลังจากดำเนินการตามขั้นตอนข้างต้น เมื่อเราเปิดโปรแกรม Database Explorer ขึ้นมาทำงาน เราจะพบชื่อฐานข้อมูลที่เราใส่ไว้ในขั้นตอนที่ 3 ปรากฏอยู่ในส่วนแสดงรายชื่อฐานข้อมูลที่โปรแกรมรู้จัก ซึ่งหมายถึงเราสามารถเขียนโปรแกรมด้วย Delphi จัดการกับฐานข้อมูลดังกล่าวได้ทันที

เรื่องของตัวอย่าง

โปรแกรมตัวอย่างที่ผมนำมาประกอบบทความในครั้งนี้ จะเป็นโปรแกรมง่าย ๆ ซึ่งจัดเก็บข้อมูลเกี่ยวกับผู้ใช้ระบบ โดยข้อมูลดังกล่าวจะจัดเก็บในตารางข้อมูลชื่อ User ตารางข้อมูลชุดนี้จะบรรจุอยู่ในฐานข้อมูลแบบ Access (Sample.mdb) อีกต่อหนึ่ง สำหรับรายละเอียดในการจัดทำโปรแกรมมีดังนี้

สร้างข้อมูลบน Access

สำหรับส่วนฐานข้อมูลแบบ Access ที่ใช้ประกอบในโปรแกรมตัวอย่างจะประกอบด้วยตารางข้อมูล (Table) ชื่อ User เพียงชุดเดียว โดยมีรายละเอียดโครงสร้างของตารางข้อมูลดังตารางที่ 1

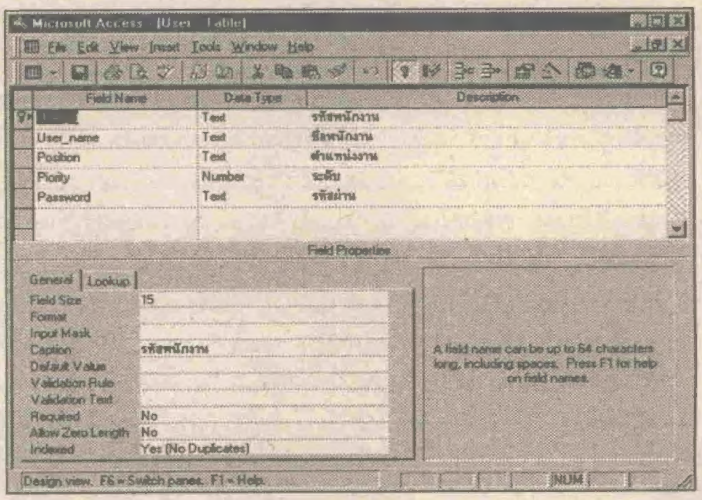
Field Name	Type	Size	Key	Other Properties
1 User_id	Text	15	*	index = "Yes Noduplicate"
2 User_Name	Text	35	-	Zero length = "Yes"
3 Position	Text	30	-	Zero length = "Yes"
4 Piority	Number (single)	2	-	
5 Password	Text	10	-	Zero length = "Yes"

ตารางที่ 1 โครงสร้างตารางข้อมูล User

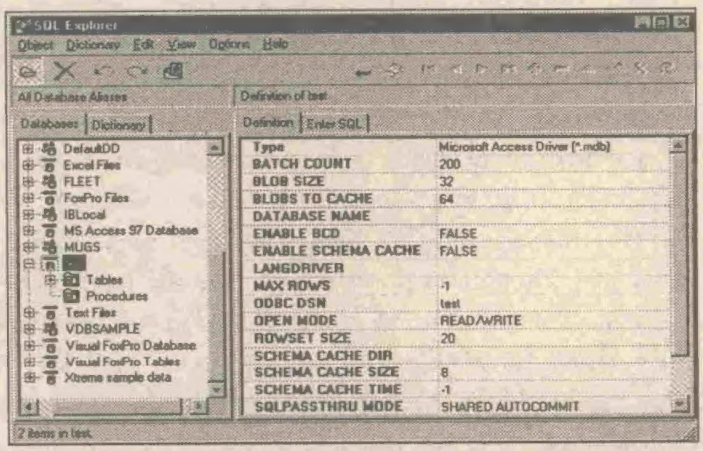
ในส่วนที่เป็นรายละเอียดเกี่ยวกับขั้นตอนการสร้างฐานข้อมูลบน MS Access นั้น ผมต้องขออนุญาตไม่อธิบายรายละเอียดใน ณ ที่นี้ เชื่อว่าท่านผู้อ่านส่วนใหญ่จะสามารถสร้างตารางข้อมูลและฐานข้อมูลขึ้นบน Access ได้ สำหรับฐานข้อมูลชุดนี้จะจัดเก็บในแฟ้มชื่อ Sample.mdb ดังที่กล่าวมาบ้างแล้วในข้างต้น

การเชื่อมโยงฐานข้อมูลแบบ Access กับ Delphi

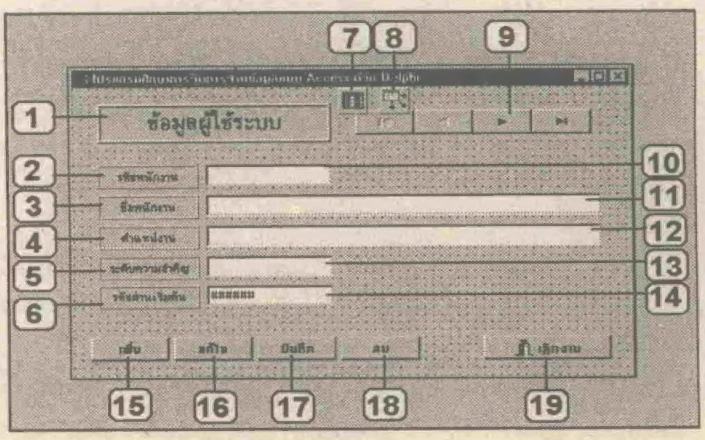
สำหรับขั้นตอนการเชื่อมโยงฐานข้อมูลแบบ Access กับ Delphi นี้ จะต้องดำเนินการบนเครื่องมือ ODBC ของวินโดวส์ ตามรายละเอียดที่กล่าวไว้ในหัวข้อก่อนหน้า สำหรับในตัวอย่างได้กำหนดชื่อฐานข้อมูลเป็น "Test" ซึ่งจะใช้เป็นชื่อเรียกหรืออ้างอิงฐานข้อมูลในการสร้างโปรแกรมใช้งานบน Delphi ในลำดับถัดไป



รูปที่ 4 การสร้างฐานข้อมูลบน MS Access



รูปที่ 5 รายชื่อฐานข้อมูลที่ Delphi รู้จัก โดยดูจาก Database Explorer



รูปที่ 6 ภาพชิ้นส่วนโปรแกรมแบบต่าง ๆ ของโปรแกรมตัวอย่าง

ตารางที่ 2 ชิ้นส่วนโปรแกรมที่จะนำมาใช้

ชิ้นส่วนหมายเลข	เป็นชิ้นส่วนแบบ	จากแถบต้นแบบชิ้นส่วน
1, 2, 3, 4, 5, 6	Panel	Standard
7	Table	DataAccess
8	DataSource	DataAccess
9	DBNavigator	DataControls
10, 11, 12, 13, 14	DBEdit	DataControls
15, 16, 17, 18	Button	Standard
19	BitBtn	Additional

ลงมือสร้างโปรแกรม

รูปที่ 6 เป็นภาพส่วนหน้าจอของโปรแกรมตัวอย่าง โดยที่บนหน้าจอจะประกอบด้วย ชิ้นส่วนโปรแกรม (Component) จำนวน 19 ชุด ซึ่งรายละเอียดของชิ้นส่วนโปรแกรมแบบต่าง ๆ ที่นำมาใช้จะอยู่ในตารางที่ 2

การปรับแต่งคุณสมบัติ (Properties) ของชิ้นส่วนโปรแกรมที่จะนำมาใช้ประกอบเป็นโปรแกรมตัวอย่างแต่ละชุด จะมีรายละเอียดตามลำดับดังตารางที่ 3

ตารางที่ 3 การปรับแต่งคุณสมบัติของชิ้นส่วนต่าง ๆ

ชิ้นส่วนโปรแกรม	หัวข้อคุณสมบัติ	ค่าที่ปรับแต่ง	หมายเหตุ
0) Form	BorderStyle	bsDialog	กำหนดรูปแบบหน้าจอ
	Caption	โปรแกรมศึกษาการจัดการฐานข้อมูลแบบ Access ด้วย Delphi	ข้อความบนแถบไตเติล
	Position	poScreenCenter	กำหนดตำแหน่งวางหน้าจอ
1) Panel1	BevelInner	bvLowered	กำหนดการเส้นกรอบใน
	BevelWidth	2	กำหนดความหนาของเส้น
	Caption	ข้อมูลผู้ใช้ระบบ	ข้อความบนชิ้นส่วน
	Font	EucrosiaUPC, Bold, 22	กำหนดรูปแบบตัวอักษร
2) Panel2	BevelInner	bvLowered	
	Caption	รหัสพนักงาน	
3) Panel3	BevelInner	bvLowered	
	Caption	ชื่อพนักงาน	
4) Panel4	BevelInner	bvLowered	
	Caption	ตำแหน่ง	
5) Panel5	BevelInner	bvLowered	
	Caption	ระดับความสำคัญ	
6) Panel6	BevelInner	bvLowered	
	Caption	รหัสผ่านเริ่มต้น	
7) Table	DatabaseName	Test	กำหนดชื่อฐานข้อมูลที่ใช้
	TableName	Table1	กำหนดชื่อตารางข้อมูล
	Action	True	เปิดตารางข้อมูล
8) DataSource1	DataSet	Table1	กำหนด DataSet
9) DBNavigator1	DataSource	DataSource1	เชื่อมโยงกับตารางข้อมูล
	ShowHint	True	ให้แสดงคำแนะนำ (Hint)
	VisibleButtons	-	เลือกปุ่มคำสั่งที่จะปรากฏ
	-nbInsert	False	ยกเลิกปุ่มคำสั่ง Insert
	-nbDelete	False	ยกเลิกปุ่มคำสั่ง Delete
	-nbEdit	False	ยกเลิกปุ่มคำสั่ง Edit
	-nbPost	False	ยกเลิกปุ่มคำสั่ง Post
	-nbCancel	False	ยกเลิกปุ่มคำสั่ง Cancel
	-nbRefresh	False	ยกเลิกปุ่มคำสั่ง Refresh

→

ตารางที่ 3 (ต่อ)

ชิ้นส่วนโปรแกรม	หัวข้อคุณสมบัติ	ค่าที่ปรับแต่ง	หมายเหตุ
10) DBEditUserID	Name	DBEditUserID	เปลี่ยนชื่อชิ้นส่วน
	DataSource	DataSource1	
	DataField	User_id	เชื่อมโยงกับฟิลด์ข้อมูล
11) DBEditUserName	Name	DBEditUserName	
	DataSource	DataSource1	
	DataField	User_name	
12) DBEditPosition	Name	DBEditPosition	
	DataSource	DataSource1	
	DataField	Position	
13) DBEditPiority	Name	DBEditPiority	
	DataSource	DataSource1	
	DataField	Piority	
14) DBEditPassword	Name	DBEditPassword	
	DataSource	DataSource1	
	DataField	Password	
	PasswordChar	#	ตัวอักษรสำหรับรหัสผ่าน
15) AddBtn	Name	AddBtn	
	Font	System	
	Caption	เพิ่ม	
16) EditBtn	Name	EditBtn	
	Font	System	
	Caption	แก้ไข	
17) PostBtn	Name	PostBtn	
	Font	System	
	Caption	บันทึก	
18) DelBtn	Name	DelBtn	
	Font	System	
	Caption	ลบ	
19) BitBtn1	Kind	bkClose	กำหนดใช้กลไกมาตรฐาน
	Caption	เลิกงาน	
	Font	System	

หมายเหตุ โปรแกรมชุดนี้จะจัดเก็บส่วนหน้าจอ (Unit) ไว้ในแฟ้มชื่อ main.pas และส่วนโปรเจ็กต์จะเก็บในแฟ้มชื่อ sample.dpr ตามลำดับ

การแทรกชุดคำสั่งควบคุมโปรแกรม

ชุดคำสั่งควบคุมกลไกการทำงานที่เราจะใส่ลงในโปรแกรมตัวอย่างชุดนี้ จะไม่ยุ่งยากซับซ้อนนัก แต่คิดว่าเพียงพอสำหรับช่วยให้ท่านผู้อ่านมองเห็นภาพการจัดการฐานข้อมูลแบบ Access ด้วย Delphi โดยจุดที่จะทำการแทรกชุดคำสั่งส่วนใหญ่จะเป็นชิ้นส่วนแบบปุ่มคำสั่ง โดยจะเลือกแทรกที่ event ชื่อ "OnClick" ของชิ้นส่วน ซึ่ง event ชุดนี้เป็น event ที่จะเกิดการทำงานเมื่อผู้ใช้คลิกที่ตัวปุ่มคำสั่ง โดยรายละเอียดของชุดคำสั่งที่จะแทรกลงในปุ่มคำสั่งต่าง ๆ มีดังนี้

- ที่ปุ่มคำสั่ง "เพิ่ม" จะทำหน้าที่เพิ่มรายการใหม่ (New Record) ลงในตารางข้อมูล ชุดคำสั่งที่ใช้จะอยู่ในบรรทัดที่ 049 ของลิสติงที่ 1

- ที่ปุ่มคำสั่ง "แก้ไข" จะทำหน้าที่กระตุ้นให้ตารางข้อมูลเข้าสู่โหมดการแก้ไขข้อมูล ชุดคำสั่งที่ใช้จะอยู่ในบรรทัดที่ 054 ของลิสติงที่ 1

ลิสติงที่ 1 MAIN.PAS

```
001 : unit main;
002 :
003 : interface
004 :
005 : uses
006 :  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
007 :  StdCtrls, Buttons, Mask, DBCtrls, Db, DBTables, ExtCtrls;
008 :
009 : type
010 :  TForm1 = class(TForm)
011 :    Panel1: TPanel;
012 :    DataSource1: TDataSource;
013 :    Panel2: TPanel;
014 :    Panel3: TPanel;
015 :    Panel4: TPanel;
016 :    Panel5: TPanel;
017 :    Panel6: TPanel;
018 :    DBEditUserID: TDBEdit;
019 :    DBEditUserName: TDBEdit;
020 :    DBEditPosition: TDBEdit;
021 :    DBEditPiority: TDBEdit;
022 :    DBEditPassword: TDBEdit;
023 :    BitBtn1: TBitBtn;
024 :    Table1: TTable;
025 :    DBNavigator1: TDBNavigator;
026 :    AddBtn: TButton;
027 :    EditBtn: TButton;
028 :    PostBtn: TButton;
029 :    DelBtn: TButton;
030 :    procedure AddBtnClick(Sender: TObject);
031 :    procedure EditBtnClick(Sender: TObject);
032 :    procedure PostBtnClick(Sender: TObject);
033 :    procedure DelBtnClick(Sender: TObject);
034 :  private
035 :     Private declarations
036 :  public
037 :     Public declarations
038 :  end;
039 :
040 : var
041 :  Form1: TForm1;
042 :
043 : implementation
044 :
045 : $R *.DFM
046 :
047 : procedure TForm1.AddBtnClick(Sender: TObject);
048 : begin
049 :  Table1.Insert;
050 : end;
051 :
052 : procedure TForm1.EditBtnClick(Sender: TObject);
053 : begin
054 :  Table1.Edit;
055 : end;
056 :
057 : procedure TForm1.PostBtnClick(Sender: TObject);
058 : begin
059 :  Table1.Post;
060 : end;
061 :
062 : procedure TForm1.DelBtnClick(Sender: TObject);
063 : begin
064 :  Table1.delete;
065 : end;
066 :
067 : end.
```

- ที่ปุ่มคำสั่ง "บันทึก" จะทำหน้าที่นำข้อมูลที่อยู่บนจอภาพบันทึกลงในตารางข้อมูลของหน่วยความจำของโปรแกรม โดยชุดคำสั่งที่ใช้จะอยู่ในบรรทัดที่ 059 ของลิสติงที่ 1

- ที่ปุ่มคำสั่ง "ลบ" จะทำหน้าที่ลบรายการข้อมูล (Delete Record) ออกจากตาราง โดยชุดคำสั่งที่ใช้จะอยู่ในบรรทัดที่ 064 ของลิสติงที่ 1

ส่วนในจุดอื่น ๆ ของโปรแกรม จะไม่ต้องแทรกชุดคำสั่งใด ๆ ไว้ ซึ่งรวมถึงที่ปุ่มคำสั่ง "เลิกงาน" ด้วย เนื่องจากในส่วนอื่น ๆ จะไม่มีการกำหนดงานพิเศษใด ๆ ไว้ ส่วนที่ปุ่มคำสั่ง "เลิกงาน" เราได้เลือกใช้กลไกพิเศษที่ Delphi เตรียมไว้ให้ (เลือกคุณสมบัติหัวข้อ Kind เป็น bkClose) ซึ่งกลไกดังกล่าวจะทำหน้าที่ปิด (Close) ส่วนต่าง ๆ ให้อัตโนมัติ

สรุป

ถ้าพิจารณาจากตัวอย่างข้างต้นจะพบว่า การจัดการฐานข้อมูลแบบอื่น ๆ ที่ไม่ใช่ฐานข้อมูลแบบ Paradox หรือฐานข้อมูลแบบ Visual dBase บนโปรแกรมที่สร้างด้วย Delphi จะไม่มีอะไรพิเศษมากนัก เพียงแต่เรากำหนดการเชื่อมโยงระหว่างฐานข้อมูลแบบอื่น ๆ กับ Delphi ให้ถูกต้องเท่านั้น การดำเนินกิจกรรมต่าง ๆ ไม่ว่าจะเป็นการเพิ่ม แก้ไข บันทึก หรือลบรายการข้อมูล จะเหมือนกับการจัดการฐานข้อมูลแบบ Paradox ซึ่งเป็นฐานข้อมูลที่มาพร้อมกับ Delphi ทุกประการ